**Вопросы по теории backend developer**

**профиль в linkedin**

**OOP**

- Объе́ктно-ориенти́рованное [программи́рование](https://ru.wikipedia.org/wiki/Программирование) ([сокр.](https://ru.wikipedia.org/wiki/Аббревиатура) ООП) - [методология программирования](https://ru.wikipedia.org/wiki/Методология_программирования), основанная на представлении программы в виде совокупности взаимодействующих [объектов](https://ru.wikipedia.org/wiki/Объект_(программирование)), каждый из которых является экземпляром определённого [класса](https://ru.wikipedia.org/wiki/Класс_(программирование)), а классы образуют иерархию наследования.

SOLID

KISS

YAGNI

**PHP**

-Трейт

- Внедрение зависимостей(здесь?)

- рекурсия

- сессии

- глобальные переменные

**SQL**

CRUD (DB, table, column, user, tuple)

Скобочки в запросе X and (Y or Z)

- Нормализация базы данных

**LARAVEL**

- Laravel/UI

-doctrine/dbal — laravel библиотека для правки полей composer require doctrine/dbal

- Аутентификация

- Авторизация

- Влидация - проверка данных на корректность на основе заданных npaвил.

- Внедрение зависимостей(здесь?)

- Постромтель запросов.

Разграничение доступа

- Посредник (middleware)- программный модуль, выполняющий предварительную обработку запроса перед запуском контроллера и (или) окончательную обработку ответа после его формирования контроллером и перед отправкой клиенту.

- Политика (policy) - программный модуль, реализующий разграничение доступа к определенной модели согласно заданным правилам.

- Провайдер (provider) - программный модуль, задающий режим работы и некоторые параметры какой-либо из подсистем фреймворка.

авторизация

аутентификация

**MVC**

**HTML**

DOM model

**CSS**

**java script**

**Apache**

**Network**

**other**

как развивался(учился)